Paper

The Template Selection in Biometric Systems Based on Binary Iris Codes

Marcin Chochowski

Biometric Laboratories, Research and Academic Computer Network NASK, Warsaw, Poland

Abstract—Since the variability of data within readings from the same person is intrinsic property of every biometric system, the problem of finding a good representative – the template – was recognized and present since the beginning of biometrics. This problem was solved differently for different biometric types, yet usually the template somehow averages the collected data samples. However, for the iris type, the template is usually just one or a few samples. In this paper we describe the experiments that suggest that the averaging is also justified in case of iris template creation. This is an important fact, which can significantly improve a performance of biometric template protection methods for iris.

Keywords—binary iris codes, biometric template selection, iris biometrics.

1. Introduction

The biometric recognition is based on comparison of the stored representative (the template) for the person in question with the newly acquired biometric sample. The resulting score of such a comparison reflects the similarity (dissimilarity) of the sample to the template. Based on a set threshold the system decides whether this score allows to state that they both originate from the same person or not. Thus it is desirable that the similarity (dissimilarity) between the selected representative and other samples from the same person is above (in case of dissimilarity below) this threshold. This requirement was sufficient for a standard biometric system to be effective.

By standard biometric system we mean a biometric system, where the decision is made upon the direct comparison of the template with the newly acquired sample. In contrast to the standard biometric system are the biometric systems that incorporate technique called the biometric template protection ([1], [2]). In those, the comparison is done not based on the similarity (dissimilarity) of the template and the sample but is an exact match between what is called *pseudonymous identifiers* generated from the template and the sample.

The pseudonymous identifier is a bit string that can be repeatable and with no errors generated from biometric data, possibly with some additional information. It is usually obtained with the help of an error correction mechanism, which might be a quantization scheme, an error-correction code or a secret sharing algorithm. For those algorithms to be efficient (to enlarge the length of the pseudonymous identifier and thus strengthen the security) it is desirable that they need to correct as few errors as possible. This yields for a template that not only will guarantee that the similarity (dissimilarity) will be above (below) some threshold, but also that the similarity (dissimilarity) between template and the samples will be as high (as low) as possible. Thus the problem of selecting the best representative as the template is restated.

2. Previous Work

The importance of selecting the best template is often underestimated. It happens that the template is simply any acquired biometric sample with no systematic procedure of its selection. In some cases there is a procedure that selects a sample that is the most similar to other samples of the same person. There are also cases where the template is created as a mean feature vector of collected samples for one person. This is well motivated by the Condorcet rule which states that an estimator (here the template) averaged over many estimators (here each code may be interpreted as an estimator of the ideal code) has smaller variance, thus is better. The question remains how to average. In this section we discuss some known approaches for template selection in different biometric.

2.1. Hand

The hand geometry biometrics uses the features that are very easy to interpret. Those features are the lengths and the widths of the fingers, the widths and the heights of the palm and other geometric features, that are gathered in one fixed-length feature vector $F = [f_1, f_2, ..., f_n]$ that takes the values from R^n. As a natural measure of dissimilarity often the Euclidean distance between such a vectors is used. It is a common practice in such a systems that the template is selected as the centroid (the mean vector) of a few samples ([3], [4]). However one must realize an important (though quite simple) fact. If we want to select the point that best represents our set in the sense that it is the closest to all the samples (it minimizes the sum of Euclidean distances between itself and other samples) than it is not the mean vector. The mean vector minimizes the *squared* Euclidean distance and it is not equivalent.

To prove this we have made an experiment with hand geometry system proposed in [4]. We have used the data set

of 149 users with at least 4 hand images each (3 of them were used to create the template and the rest for comparisons). For every user two templates were created – the first one as the mean vector of 3 sample and the second as on of the 3 samples that was closest to other two. For those two templates we calculated genuine and impostor scores (resulting in 179 genuine scores and 45105 impostor ones) using two dissimilarity measures – Euclidean distance (*Euc*) and squared Euclidean distance (Euc^2). The results are compared on the basis of the equal error rate (EER) (Table 1) – this maybe a bit simplifying, though it shows an important fact. The method of template selection should be adjusted to dissimilarity/similarity measure, in particular a mean vector is not an appropriate template when using Euclidean distance. For Euclidean distance the best template out of 3 gives better results and for squared Euclidean distance the mean code performs better.

Table 1
EER results for different configurations of template and dissimilarity measure

EER	*Euc*	Euc^2
Best [%]	**7.48**	7.81
Mean [%]	7.56	**7.24**

2.2. Fingerprint

There was much research put into the feature extraction and matching algorithms for fingerprint minutiae but respectively little attention (as in other biometric modalities) was given to the problem of template selection. There were some analysis of different selection of representative fingerprint impression that either best represents the intra-class variations or maximizes the similarity with the rest of the impressions [5]. The results showed that a systematic template selection is much better than random selection. Further work on template creation for fingerprint showed that it is reasonable not to choose a single impression but merge few impressions (mosaicing) of the same fingerprint resulting in bigger coverage of the finger thus better representation.

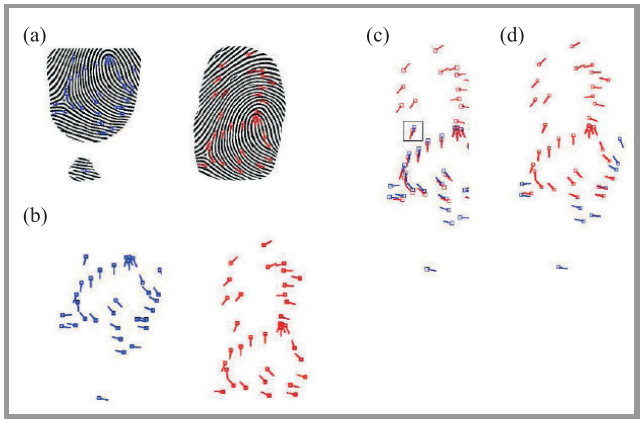

Fig. 1. Fingerprint features mosaicing, (a) two impressions, (b) minutiae extracted from impressions, (c) alignment, (d) mosaicked template [6].

In [6] Ross *et al.* analyzed three different techniques of data merging. The firs was mosaicing on the image level. They aligned the images and merged them using thin plate splines, and then extracted minutiae and performed matching using those minutiae as template. The second approach was to first extract the minutia form two impressions and do the mosaicing on the minutiae level and use the merged minutiae as the template (Fig. 1). The third method was to separately use both impressions (matching two minutiae representations) and fusing the matching scores. The experiments showed that the second method (mosaicing on minutiae level) gives the best results and outperforms single impression matching.

These results are especially important for biometric cryptography (template protection) methods. Most of them that use the fingerprint use the fuzzy vault algorithm (see, e.g., [7]) where a good coverage of fingerprint is one of the most important aspects. This was showed by Nandakumar in his implementation of fuzzy vault for fingerprints ([8]). The usage of mosaiced template improved the results for genuine acceptance by as much as 4% not decreasing the security (false acceptance).

2.3. Signature

Some recent findings in the area of handwritten signatures based on the theory of warped least squares, prove that an template called *the hidden signature* can be defined that greatly improves the performance of matching. This hidden signature can be interpreted as a mean template, but the averaging is done in warped space – for any signature a transform (a warping path) is defined that map it to the space of the warped template were the comparison is done. See [9] for details.

2.4. Iris

As for the iris biometrics, there is no common methodology for template selection. In most cases the template is simply an iris code of acquired image ([10], [11]), or a set of iris codes ([12]). Sometimes like in BiomIris ([13]) the template is chosen as one out of three codes, such that it minimizes the sum of distances to two others. There were also suggestions that average code created as the majority code could be a better representation [14] however that has been argued to have limited use in practice ([15]).

We have to also keep in mind that there are different coding methods for iris recognition that end up with binary code. The question is whether the selection method the best representative would be the same for different algorithms or rather it is algorithm-specific. To address this we propose a few different candidates for the template and verify their effectiveness for two different coding algorithms, namely the OSIRIS implementation of Daugman coding [10] and Czajka's algorithm [16]. OSIRIS is an implementation of

Daugman-like iris texture coding. In our particular realization it produces an binary iris code of length 1974 bits. The Czajka's algorithm represents a different approach to texture coding using Zak-Gabor transform. It produces a binary iris code of length 1024 bits.

3. Selection Methods

3.1. Notation

Let us define the following notation that will be valid hereafter:

- $I = \{0,1\}^N$ – space of binary codes of length N (vertices of a unit hyper-cube),
- $A \subset I, A = a_1, a_2, \ldots, a_K$ – set of K available iris codes for particular person, a – iris code,
- a_{ij} – j^{th} bit of i^{th} code for the same person, $i = 1,\ldots,K,\ j = 1,\ldots,N$.

For simplicity, to omit unresolved cases let us assume that the K is odd.

3.2. Possible Candidates

Now we can define different candidates for the iris template. Let us define the average code as

$$\bar{a} = \left(\frac{1}{K}\sum_{i=1}^{K} a_i\right),\ \bar{a} \in R^N,$$

where a_i is an N-dimensional iris code $i = 1 \ldots K$.
We can write also,

$$\bar{a} = \arg_{a' \in R^N} \min \sum_{a \in A} \left\|a - a'\right\|^2.$$

This follows from the fact, that the second moment is minimal around the mean value, thus we interpret the code $\bar{a}$ as the real code that minimizes the squared Euclidean distances from all codes from set A – best *represents* them. Let us also define the majority code as

$$a_I^M = \left(\mathbf{Maj}\left(\sum_{i=2}^{K} a_{ij} - \frac{N}{2}\right),\ j = 1,\ldots,N\right),\ a_I^M \in I.$$

This is the code that has jth bit equal to 1 if among K codes there were more 1's than 0's on this position and 0 otherwise. Since we assumed K to be odd we excluded the case in which the number of 1's and 0's is equal.
The relation between the code $\bar{a}$ and a_I^M is summarized by the following theorem.

Theorem 1: The majority code a_I^M is the nearest code from the subspace I to the average code $\bar{a}$.

Proof: The relation between the mean and the median is as follows:

$$\begin{aligned} |m - m_e| = |E(X - m_e)| &\leq E(|X - m_e|), && (1)\\ &\leq E(|X - m|), && (2)\\ &= E(\sqrt{(X-m)^2}),\\ &\leq \sqrt{E((X-m)^2)}, && (3)\\ &= \sigma. \end{aligned}$$

The Eq. (1) inequality comes from the property of sum of absolute values, the Eq. (2) inequality comes from the fact that the median value minimizes the absolute deviation function. The Eq. (3) inequality comes from the Jensen's inequality, for the concave functions (square root function).
Thus the mean m value is less than σ from the median m_e.

$$|m - m_e| < \sigma$$

what proves the theorem. ■

Yet we know that under taken assumptions (K is odd) we have $\sigma < 0.5$ and $m \in <0,1> - \{0.5\}$. That means that the median code is the closest binary code ($\in I$) to the average code.

$$a_I^M = \arg_{a' \in I} \min \left\|a' - \bar{a}\right\|^2$$

At the same time, from the properties of median, we have

$$a_I^M = \arg_{a' \in I} \min \sum_{a \in A} \left|a - a'\right|.$$

Since $|.|$ and $\|.\|^2$ are equal for the subspace I, we see that the a_I^M is an analog of the average code but with constraints to the solution space.

$$a_I^M = \arg_{a' \in I} \min \sum_{a \in A} \left\|a - a'\right\|^2$$

We can also point out two additional codes from the set A. The code that is closest to the average code and the code that is closest to majority code. Those are defined as respectively

$$\bar{a}_A = \arg_{a' \in A} \min \left\|a' - \bar{a}\right\|^2 \quad \text{or} \quad (\bar{a}_A = \arg_{a' \in A} \min |a' - \bar{a}|),$$

$$a_A^M = \arg_{a' \in A} \min \left\|a' - a_I^M\right\|^2.$$

There is also a code often used as the template that is defined as

$$a_A^T = \arg_{a' \in A} \min \sum_{a \in A} \left\|a - a'\right\|^2$$

and is the analog of the majority binary code but selected from the set A (set of known sample codes). This is the code previously called the best code.
Intuitively the best representation, contrary to [15], would be the majority code.

4. Experiments

According to above discussion we performed a series of experiments to verify the usability of different template selection. In the following experiments we have used part of BATH database (110 eyes with 20 images per eye). For each experiment always the first 10 images were used to create the template and the rest 10 were used as samples for comparisons. The genuine comparisons were performed with the template against 10 genuine samples what makes $1100 = 110 \cdot 10$ comparison in total, and impostors with the template against 10 samples of all other eyes what makes $119900 = 110 \cdot (110-1) \cdot 10$. The experiments were performed for two coding methods - OSIRIS and Czajka's coding. To describe the results we calculated several parameters including false non-match rate (FNMR), false match rate (FMR) and, as suggested in [17], decidability index d'. The FNMR was calculated as the rate of positive samples wrongly classified as negative ones, FMR as the rate of negative samples wrongly classified as positive ones and EER as the rate where FNMR and FMR are equal.

4.1. OSIRIS Coding

First we wanted to compare the performance of the recognition algorithms depending on the way the template is created. In particular we compared the performance using the majority code defined as a_I^M the best iris code defined as a_I^T and iris code that is the closest to the majority code a_A^M. To compute the majority code we have aligned normalized iris images (in polar format) using 2D correlation, compute the codes for each image and took the median value for each code bit (although the number of samples was even none of the bits for all codes was 0.5). To select the best code we have cross-matched all the 10 codes and selected the one that had the minimal sum of distances to the rest 9.

Next, for different templates, we performed the verification according to the protocol defined above. Figure 2 plots the cumulative distributions (we do not plot the histograms for clarity) of genuine and impostor comparisons for different template selection method. We see that the differences are significant and the best results were obtained for majority code – Table 2 summarizes the results. With the majority code we obtained the perfect separation and good decidability index. Additionally there are plotted results obtained when using as the template one of the 10 codes selected at random labeled as *random iris code*.

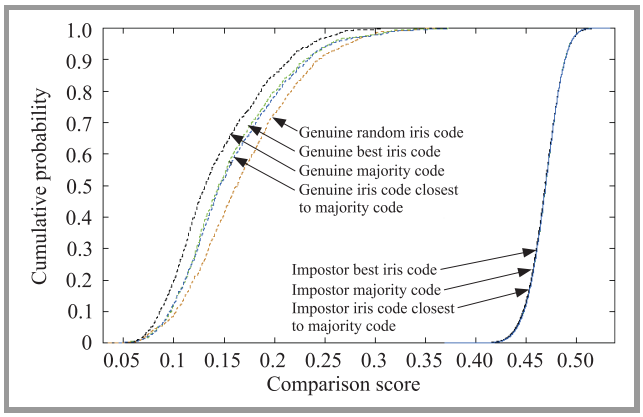

Fig. 2. Cumulative distributions of genuine and impostors scores for different templates (best code, random code, majority code and code closest to majority) for OSIRIS coding algorithm.

Table 2
Summary of verification performance for different template selection method for OSIRIS coding algorithm

Indexes	Best code	Majority code	Closest to majority code	Mean code
EER [%]	0.0017	**0**	0.0017	0
FNMR (FMR = 0%) [%]	0.27	**0**	0.18	0
d'	7.82	**8.84**	7.62	9.06

To analyze the averaging property we decided to compare those results with two more possibilities of average template – namely $\bar{a}_A$ (iris code closest to the real-value average) and $\bar{a'}_A$ (iris code closest to the real-value average in L_1-norm). The results are plotted in the Fig. 3. Still the majority code outperforms the others, but surprisingly the code $\bar{a}_A$ is as bad as randomly selected code whereas the code $\bar{a'}_A$ is as good as the best code a_A^T and closest to majority code a_A^M. There is one more very interesting property worth noticing. The methods that selected one of the iris codes as the template did select different codes thus we cannot infer that some of presented methods are equal. Intuitively we guess that, e.g., a_A^M should be the same as a_A^T, but that is not the case.

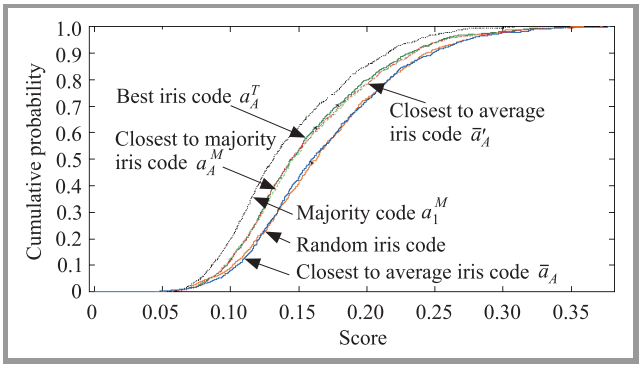

Fig. 3. Genuine comparisons scores (normalized hamming distance) cumulative distributions for different template selection methods

The above described experiment with use of the OSIRIS as the coding algorithm proved the assumption that creating the template code by averaging leads to better performance. The difference between the performance is significant. It is clear that the majority code gives the best results whereas the results with the best code as the template are much worse and with the random code are the worst. This proves

the statement of Davida ([14], the averaging by majority coding as the template creation has influence on the performance and gives much better results.

Noting this fact we decided to go one step further and represent the template as a real-value vector $T_{pr} = [p_1..p_{1974}]$ in which each position p_i represents the rate of this bit was equal one in codes used to create the template. Thus it is a vector with elements from $< 0, 1 >$ that could be interpreted as probabilities of 1 on that position in iris code of particular person. This of course makes the template much bigger since it is no longer represented as N bits, but N real numbers, yet the size of it (precision) depends on the number of samples used for template creation and could not be very high. Nevertheless a more complicated problem is the matching algorithm. We can no longer use the ExOR operation and other method should be proposed.

A natural selection of a distance measure is the squared Euclidean distance. The sample codes are the vertices of the 1974-dimension hyper-cube and the templates are points inside this cube. The similarity measure is simply the distance between a vertex and that point. The results on the same data as before with this methodology gives very good results. We obtain perfect separability with decent-looking histograms (Fig. 4).

Yet there could be another similarity measure. If we would extend the p_i to function P_i so that it is a probability function for ith position in the code such that $P_i(x = 1) = p_i$ and $P_i(x = 0) = 1 - p_i$ a natural method to verify a new code could be measuring how probable is it, given the template – simply multiply. Of course the probabilities for the elements in the vector are dependent thus multiplying them is not theoretically justified, but the experience in machine learning lets us expect reasonable results. Additionally we have to guarantee that there will be no 0 probabilities to eliminate the effect of zeroing the score (each template element with 0 value, meaning that for all codes used to create the template that particular bit was always 0, was set arbitrary to 0.01). Since there is 1974 bits in the code, calculation of the *pseudo-probability score* (Pr_{score}) by multiplying subsequent values is numerically difficult thus we applied log operation and summed the logarithms.

$$Pr_{score}(T_{pr}, a) = \prod_{i=1\ldots1974} (P_i(x{=}a_i)) = \exp\left(\sum_{i=1}^{1974} \log_e(P_i(x{=}a_i))\right),$$

where T_{pr} is the template with probabilities functions P_i for respective positions $= 1 \ldots 1974$. Unfortunately the obtained values of Pr_{score} were of form $\exp(k)$ where k for genuine comparisons was about minus few hundreds

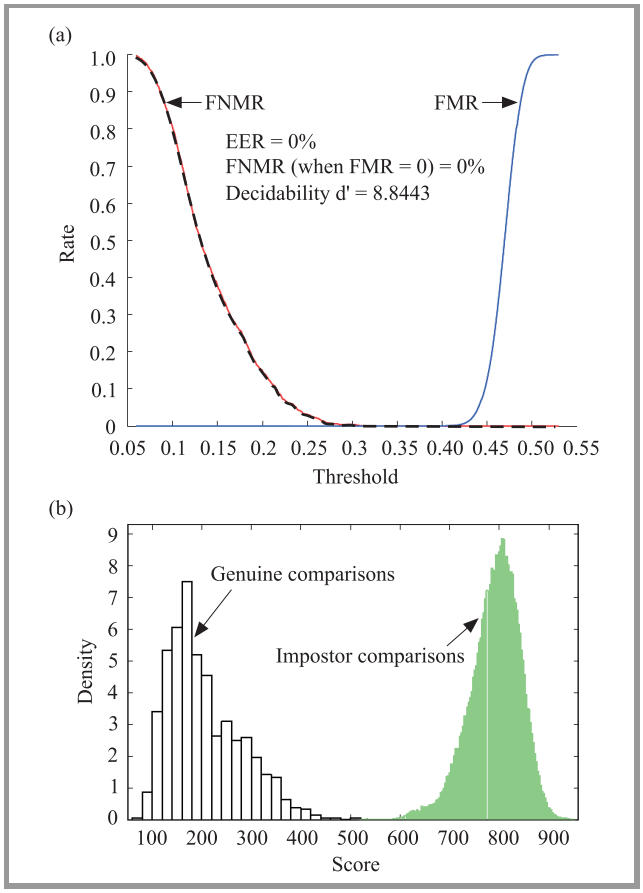

Fig. 4. (a) FNMR and FMR graphs, (b) performance rates (left) and comparisons histograms (right) for OSIRIS system with real-valued mean template code and Euclidean distance as similarity measure.

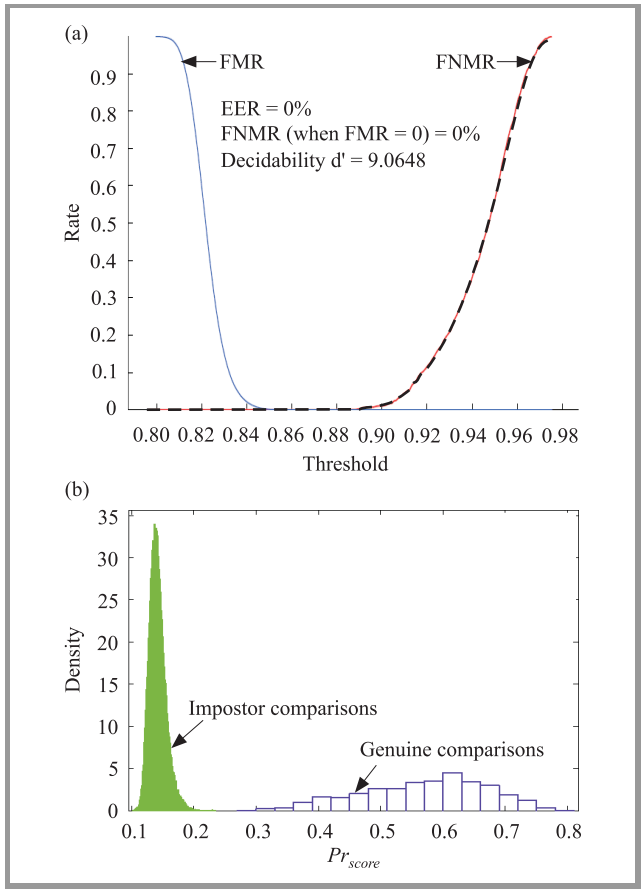

Fig. 5. (a) FNMR and FMR graphs, (b) performance rates (left) and comparisons histograms (right) for OSIRIS system with real-valued mean template code and modified matching algorithm.

and for impostors minus few thousands. Thus we decided to divide the k by 1000 to observe the matcher performance. Figure 5 plots the FMR, FNMR for this matcher ($P_{score} = \exp(\sum_{i=1}^{1974} \log_e(p_i)/1000)$).

The results are very promising. With this approach we obtain full separation and the histogram shapes (which have their reflection in d') indicate that such an approach is reasonable and may lead to better results than standard one.

To look into the influence of the template selection on the performance of the iris biometric system we observed also plots representing so-called Dodington-zoo menagerie. This is very helpful for security analysis. It shows whether all irises (with respect to coding algorithm) are equally different or are there some types of irises that either are more similar to others or are less similar to itself. Recent paper from Yager and Dunstone [18] introduced new division and naming for different *behavior* of biometric data depending on mean impostor and genuine scores. We do not want to go into the details of deciding what is a normal behavior and what is not. Instead we want to know whether the different template selection algorithm influences this behavior. Figure 6 plots the menagerie plots for scores obtained with

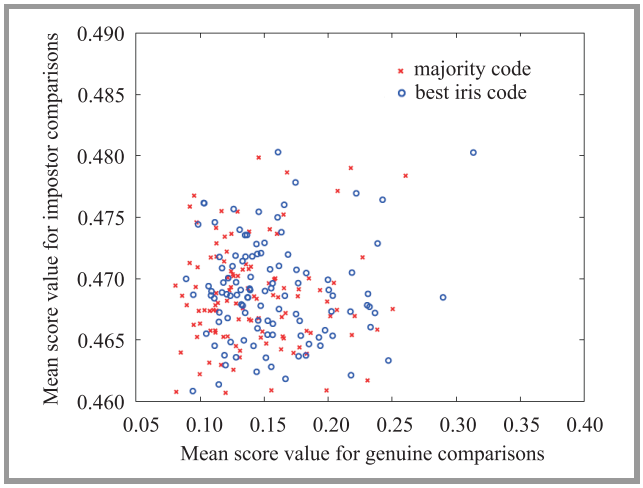

Fig. 6. Menagerie plot for two different template selection methods using OSIRIS coding. Each point represents a single iris showing how well it is on average matched to itself and other irises.

best iris code (circles) and majority code (crosses) as the template. It shows that there is no influence as for the mean impostors scores, what is a good property, and the rightmost mean genuine comparisons for majority code are much smaller what is even better property.

4.2. Czajka's coding

We performed similar experiments using Czajka's coding. Again comparing the genuine cumulative distributions we noticed that the majority code outperforms others giving the best results (Fig. 7). However the behavior of others is significantly different than in case of OSIRIS coding. Here we see that the iris code closest to the majority code gives very poor results (almost as bad as randomly selected iris

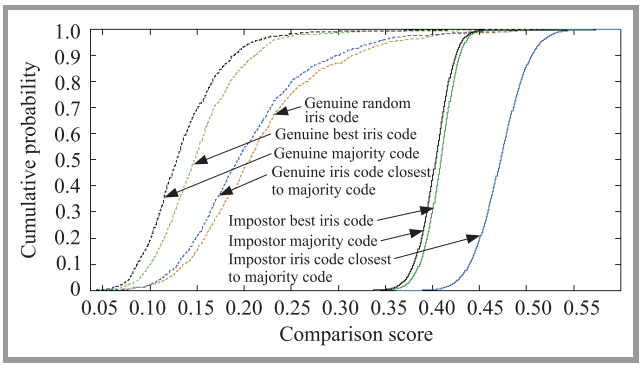

Fig. 7. Cumulative distributions of genuine and impostors scores for different templates (best code, random code, majority code and code closest to majority) for Czajka's coding algorithm.

code). This lets us suspect that the codes created by this algorithm are oddly distributed in the code space, since although majority code estimates the codes well the nearest code does not. Perhaps in this case the majority code does refer any real iris image, but is rather an *virtual object*. The behavior of impostor distributions is even more wired. For OSIRIS, there were no differences for different templates, and here the differences are very significant. We see that the impostor comparisons with majority code are slightly worse (give lower dissimilarity score) and that choosing bad template (iris code closest to majority code gives poor genuine scores) can move the impostors to the right. We can guess that the first observation may be due to not equal distributions of ones and zeros in this type of coding thus averaging may lead to code that better fits different codes (e.g., has more ones). The second observation results from the fact that a bad template in more *noisy*, hence the impostor scores look more random. Both of these facts may prove that this type of coding codes not only the individual characteristics but also some kind of *more global* information. This is quite interesting conclusion and will be a subject for further research.

Table 3 summarizes results of the experiments. We see that we obtained worse results than for OSIRIS but the averaging property of majority code is visible also in this case.

The cumulative distributions from Fig. 7 let us assume that we may expect undesirable distribution changes in menagerie plots. Indeed, Fig. 8 shows that changing the template selection method for majority coding the users tend to be more *wolfy* – mean value of impostor scores gets smaller (different users are more similar).

Table 3
Summary of verification performance for different template selection method for Czajka's coding algorithm

Indexes	Best code	Majority code	Closest to majority code
EER [%]	0.662	**0.542**	1.60
FNMR (FMR = 0%) [%]	0.82	**0.63**	3.09
d'	6.87	**7.69**	4.99

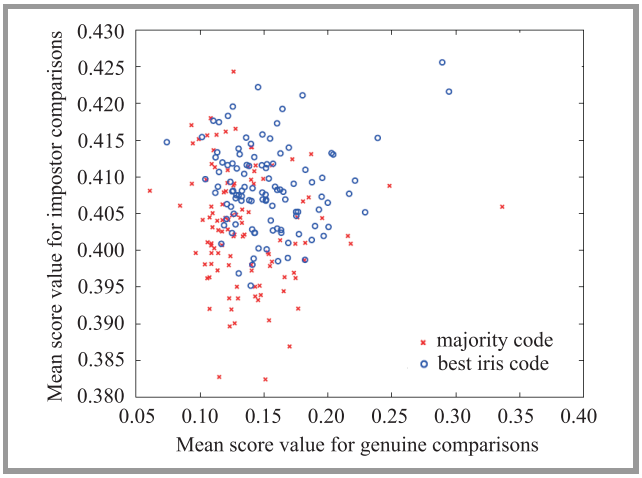

Fig. 8. Menagerie plot for two different template selection methods using Czajka's coding. Each point represents a single iris showing how well it is on average matched to itself and other irises

These experiments prove that a really good understanding of the codes and their properties is needed to propose an bio-encryption algorithm for it.

5. Conclusions and further work

Concluding these experiments we claim (in opposite to other authors, e.g., [15]) that for binary iris coding algorithms using the majority code as the template leads to better results. These experiment prove how important is the template selection problem. It was not addressed before in work on biometric template protection, but it seems to be crucial for most of the methods used there. All of them assume that we have a reference code that can be seen as a codeword of error-correcting code and all the query codes lie around it in a distance less than assumed threshold. We showed that depending on the template selection we can obtain different results and that the good understanding of the space of the codes is crucial.

Acknowledgements

This paper has been financed by the Ministry of Science and Higher Education grant OR00 0026 07 "*A platform of secure biometrics implementations in personal verification and identification*".

References

[1] ISO/IEC, "CD 24745.2 Information technology – Security techniques – Biometric template protection".

[2] A. K. Jain, K. Nandakumar and A. Nagar, "Biometric template security", *EURASIP J. Adv. Sign. Proces., Spec. Issue Adv. Sign. Proces. Patt. Recogn. Meth. Biometr.*, pp. 1–17, Jan. 2008.

[3] R. Sanchez-Reillo and C. Sanchez-Avila, "Biometric identification through hand geometry measurements", *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1168–1171, 2000.

[4] L. Stasiak, "Weryfikacja tożsamości poprzez wykorzystanie cech dłoni", M.Sc. thesis, Institute of Control and Computation Engineering, Warsaw University of Technology, Warsaw, 2004.

[5] U. Uludag, A. Ross, and A. K. Jain, "Biometric template selection and update: a case study in fingerprints", *Patt. Recogn.*, vol. 37, no. 7, pp. 1533–1542, 2004.

[6] A. Ross, S. Shah, and J. Shah, "Image versus feature mosaicing: a case study in fingerprints", in *Proc. SPIE, Biometric Technologies for Human Identification III*, Orlando, USA, 2006, vol. 6202, pp. 1–12.

[7] A. Juels and M. Sudan, "A fuzzy vault scheme", in *IEEE Int. Symp. Infor. Theory*, Lausanne, Switzerland, 2002, pp. 408–421.

[8] K. Nandakumar, A. K. Jain and S. Pankanti, "Fingerprint-based fuzzy vault: implementation and performance", *IEEE Trans. Inform. Foren. Secur.*, issue 4, vol. 2, pp. 744–757, 2007.

[9] J. Putz-Leszczyńska and A. Pacut, "Hidden signature – a new solution for on-line verification using DTW", in *Proc. 42nd Ann. IEEE Int.l Carnahan Conf. Secur. Technol. ICCST 2008*, Prague, Czech Republic, 2008, pp. 162–166.

[10] E. Krichen, A. Mellakh, S. Salicetti, and B. Dorizzi, "OSIRIS (Open Source for IRIS) reference system", *BioSecure Project* [Online]. Available: http://www.cilab.upf.edu/biosecure1/public docs deli/BioSecure Deliverable D02-2-2 b4.pdf.pdf

[11] L. Masek and P. Kovesi, *MATLAB Source Code for a Biometric Identification System Based on Iris Patterns*. The School of Computer Science and Software Engineering, The University of Western Australia, 2003.

[12] "VeriEye 2.2 SDK" [Online]. Available: www.neurotechnology.com

[13] "BiomIris SDK" [Online]. Available: www.BiometricLabs.pl

[14] G. I. Davida, Y. Frankel, and B. J. Matt, "On enabling secure applications through off-line biometric identification", in *Proc. IEEE Symp. Secur. Priv. 1998*, Oakland, USA, 1998, pp. 148–157, 1998.

[15] F. Hao, R. Anderson, and J. Daugman, "Combining cryptography with biometrics effectively", Techn. Rep. UCAM-CL-TR-640, University of Cambridge, 2005.

[16] A. Pacut, A. Czajka, and P. Strzelczyk, "Iris biometrics for secure remote access", in *Cyberspace Security and Defense: Research Issues*, J. S. Kowalik *et al.*, Eds. Dordrecht: Springer, 2005, pp. 259–278.

[17] J. Daugman, "High confidence personal identification by rapid video analysis of iris texture", in *Proc. IEEE Int. Carnahan Conf. Secur. Technol.*, Atlanta USA, 1992, pp. 50–60.

[18] N. Yager and T. Dunstone, "The Biometric Menagerie", *IEEE Trans. Patt. Anal. Mach. Intell.*, vol. 32, no. 2, pp. 220–230, 2010.

Marcin Chochowski received his M.Sc. in 2004 from the Faculty of Electronics and Information Technology of the Warsaw University of Technology. He is presently a Ph.D. student at the Institute of Control and Computation Engineering at the Faculty of Electronics and Information Technology of the Warsaw University of Technology. Since 2003, he is also a research assistant at the Biometric Laboratory of Research and Academic Computer Network NASK. He is interested security aspects of biometric systems, artificial intelligence, statistics and related areas.

e-mail: Marcin.Chochowski@nask.pl
Research and Academic Computer Network (NASK)
Biometric Laboratories
Wąwozowa st 18
02-796 Warsaw, Poland